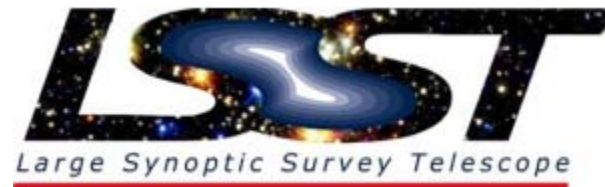

Real-life Data Intensive Applications – Challenges and Solutions

Jacek Becla
SLAC National Accelerator Laboratory

2010 Salishan Conference on High Speed Computing

My World....



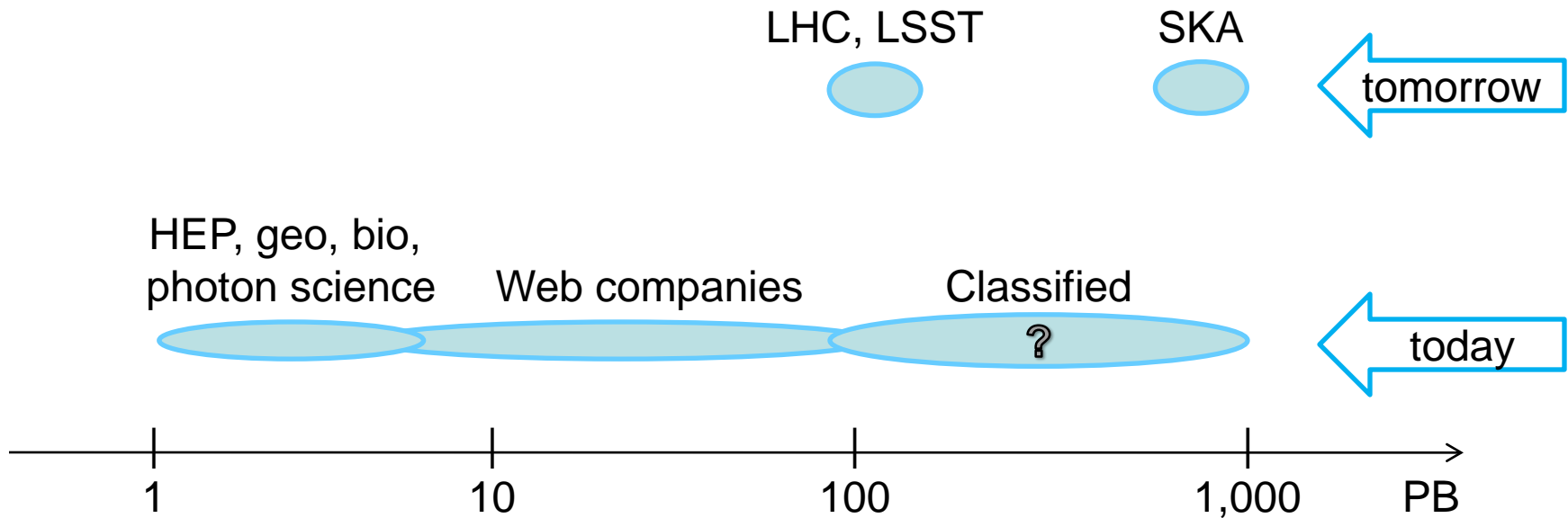
Focus

- Complex **analyses**
on **observational, scientific** data
- Practical solutions
- Extreme scale (think: 100+ PBs)

Outline

- **Extreme scale scientific analyses**
- Data intensive computing realm
- Complexity of scientific data sets
- Current trends
- Existing solutions
- Summary

Extreme Scale



Scientific Analyses

- Needle in haystack
 - Unsure what the needle looks like
- Time series
- Spatial correlations
- Real-time outliers detection

Outline

- Extreme scale scientific computing
- **Data intensive computing realm**
- Complexity of scientific data sets
- Current trends
- Existing solutions
- Summary

Bandwidth, not Capacity

- 1 PB @50MB/sec = 230 days
- 1 PB in 1h @50MB/sec/disk → 6K disks
 - but 1TB disk not uncommon today
- **I/O driven**, not capacity driven
 - Multiple copies often come for “free”
- Can trade some I/O for CPU
 - Compute on the fly
 - Compress (so-so for science data)

Big Bandwidth → Big Clusters

- Too many disks/node
= memory bottleneck
- Clusters measured in 100s, 1,000s
- Challenge
 - Mgmt overhead, full automation
 - Dealing with routine failures
 - MTBF= 50years & 6K disks = failure every 3 days
 - Avoiding shared resources

Petabyte > One Table

- Data must be partitioned and distributed
 - Many trade-offs!
- Many reasons
 - Petabyte in a single table not an option
 - Large projects = distributed funding/computing
 - Distributing for backup
 - Specialized data centers

Issues in Partitioning & Distribution

- Large #partitions vs large partitions
- Fixed-size vs variable-size chunking
- Progressive
- Adaptive
- 1-level vs 2-level partitioning
- Materialized vs on-the-fly
- Overlaps
- Random vs controlled distribution

Constant Change → Flexibility

- Grow incrementally
 - Scale out
- Uncertainty, highly varying load
 - System has to adapt, don't want to overbuild
- Large monolithic systems are hard to make failure proof
 - Complexity in H/W vs in S/W

Other Challenges

- Cost estimate
- Approx results
 - to speed up exploration
 - to skip failed nodes (if acceptable)
- Job pause/restart
- Self management
 - auto-load balance, auto-fail over, auto-QA
- Relaxed consistency
- Provenance tracking

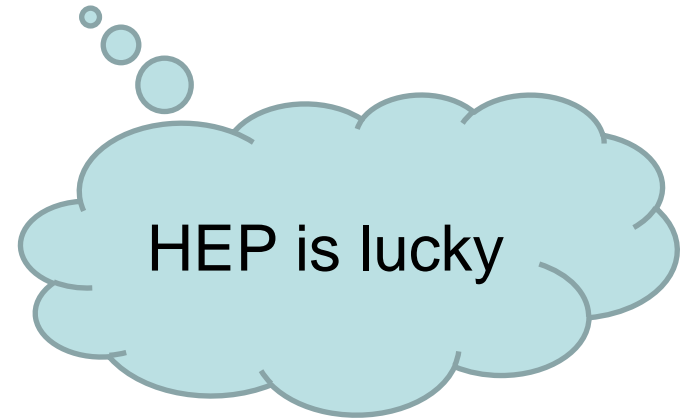
Outline

- Extreme scale scientific computing
- Data intensive computing realm
- **Complexity of scientific data sets**
- Current trends
- Existing solutions
- Summary

Data is Clustering-Intensive

- Order
 - Time series
- Locality
 - Spatial, temporal
- Adjacency
 - Neighbors
- Correlations
 - Densities
 - External catalogs

This applies to
many sciences...
geo, astro, bio



...Multi-Dimensional and Uncertain

- Typically few dimensions
 - Spatial (2-3)
 - Temporal
 - Sometimes frequency
- Can't effectively cluster on all dimensions
- Uncertain
 - Measurements
 - Results

Many Industries Are No Different

- Weblog analytics
 - Personalization of rankings using predictive modeling
 - Netflix \$1M challenge
 - Optimizing ad placement
 - What-if analysis to tune search engines
- Financial services
 - Risk calculation; risk management
 - Long term strategy modeling
 - Real-time trading models
- Deep sequencing analytics for drug discovery
 - Put whole gene together from overlapping fragments where each segment carries probability of correct decode
- Digital medical imaging analytics
 - Find all the patients with MRIs that looked like this one
- Oil and gas discovery geological data
 - Produce an underground map from signal data

Outline

- Extreme scale scientific computing
- Data intensive computing realm
- Complexity of scientific data sets
- **Current trends**
- Existing solutions
- Summary

Scientific Analytics – Paradigm Shift

- Do-it-yourself analyses do not scale
 - Petabyte won't fit on your laptop
- Extreme analyses requires centralization
 - Data providers vs data analyses centers
 - Moving computation to data and sharing resources much more cost effective at extreme scale
- Application specific optimizations

Shared Nothing Clusters

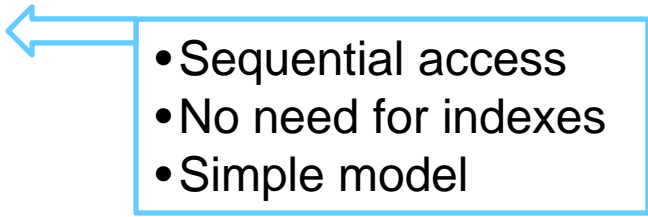
- Why not SAN?
 - Designed for management not bandwidth
 - Expensive
 - Inter-switch bandwidth limits
 - No fine-grain control over placement
 - Sending data to query
- Why not traditional HPC?
 - Designed for FLOPS not I/O
 - Assumes little data movement

Pushing Computation to Data

- Moving data is expensive
- Push computation to data or compute “near” data
- Happens at every level
 - Send query to closest center
 - Process query on the server that holds data

I/O and Network Improvements

- Limit accessed data
 - Generate commonly accessed data sets
 - Columnar stores
- De-randomize I/O
 - Copy and re-cluster pieces accessed together
- Segregate and combine I/O
 - Separate random reads from sequential scans
 - Tune placements and indexing per data set
 - Share scans
- Trade CPU for I/O

- 
- Sequential access
 - No need for indexes
 - Simple model

Outline

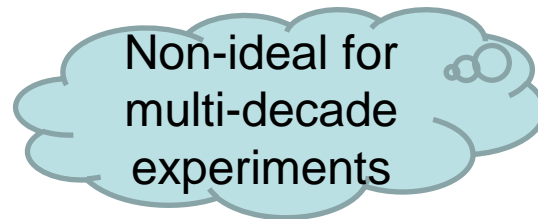
- Extreme scale scientific computing
- Data intensive computing realm
- Complexity of scientific data sets
- Current trends
- **Existing solutions**
- Summary

Data Mgmt Systems in Practice

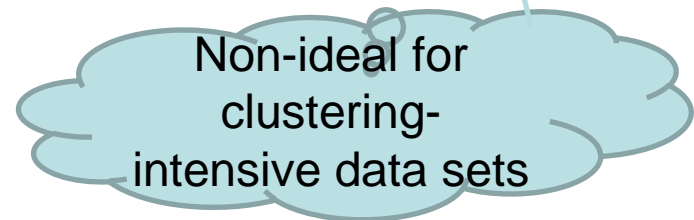
- Off-the-shelf RDBMS based
 - eBay, WalMart, Nokia, BaBar, SDSS, PanSTARRS, LSST
- Custom software, structured files + metadata in RDBMS
 - All HEP, most geo, many in bio, ...
- Custom software, custom format
 - Google, Yahoo!, Facebook, ...
(but still use RDBMS for OLTP)

DBMS vs Hadoop & Map/Reduce

- System catalog and storage manager
 - Knows where relevant data resides
 - Co-locates related sub-regions
- Processing close to the data
- No Underlying data and storage model
 - Schema in application code
 - Data hash partitioned
- Processing near the data (akin to ETL)



Non-ideal for multi-decade experiments



Non-ideal for clustering-intensive data sets

Convergence

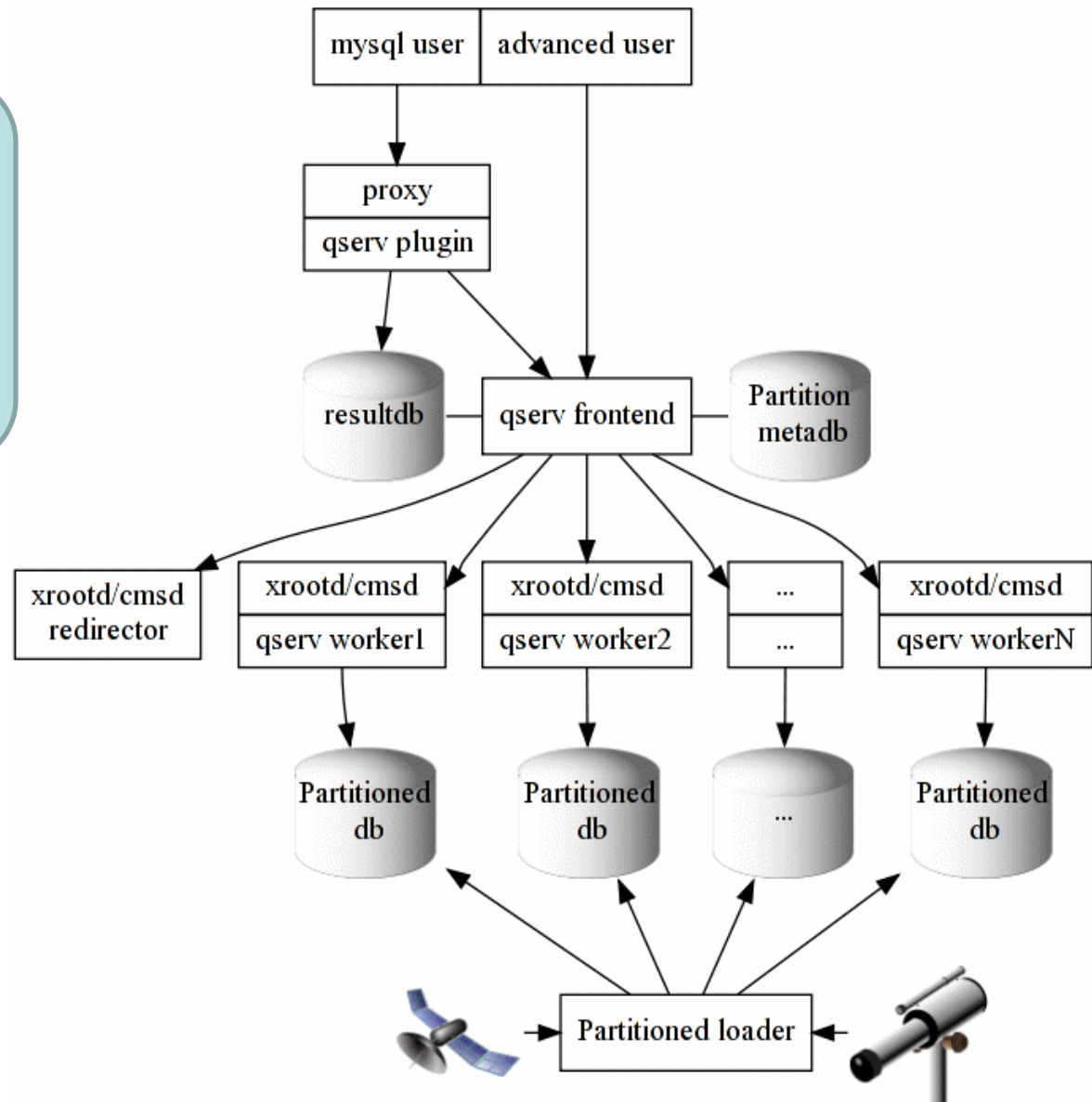
- DBMS vendors
 - Rush towards shared-nothing*
 - Teradata had it, IBM: DB2 Parallel Edition, Oracle: Exadata, Microsoft: Madison
 - Emergence of shared-nothing MPP DBMS startups
 - Adding map/reduce paradigm support
 - AsterData, Greenplum, Teradata, Netezza, Vertica
- Map/Reduce
 - Rush to add db-ish features (schemas, indexes, more operators)

Query Service (qserv)

- Shared-nothing on top of MySQL
- Built for analyses on immutable data sets
- Optimized for spatial and temporal analyses on extreme scale data sets
- Overlapping partitioning, fixed chunks, 1st level materialized, 2nd on the fly
- Shared scans (available ~Q4'10)
- Fault tolerance
- Usable prototype in public domain in Q2'10

Qserv Architecture

Deploying for wide use by LSST science collaborations on 20 TB data set this year



SciDB

Open source DBMS for
scientific research

- Shared-nothing MPP DBMS
- Arrays
 - natively supported arrays
(basic, enhanced: ragged, nested...)
 - array operators

Traditional RDBMS vs Arrays

- Data model
 - Need n-d arrays, not tables
 - Simulating arrays on top of tables costs $\sim x100$
 - Locality, adjacency is natural in n-dimensional space
 - Tracking uncertainty or units becomes just another dimension
- Operations
 - Need array operators and parallel user-defined-functions not SQL
 - Think regrid, smooth, not join



SciDB (...cont)

- Overlapping partitions
- Basic uncertainty support
- Scalability to 100s PB, 1,000s nodes
 - high degree of tolerance for failures
- Massively parallel system, including user defined functions
- AQL (an array & analytics query language)
- Extensibility for integrating domain specific algorithms, languages or packages like R and MATLAB
- In-situ data, including netCDF and HDF5
- Named versions
- Shared scans
 - Attribute-store with aggressive compression (multiple options)

SciDB (...cont)

- Ideal for...
 - Managing / analyzing gridded / n-d data sets
 - Such as images
 - Complex analyses on large data sets
 - Time series, spatial correlations, curve fitting, eigenvalues, covariance
- Strong team
 - 20+, including world-class database pioneers
 - Mostly volunteers from academic, science and industrial communities
- 3 POC's underway
 - LSST (demo @VLDB), quantitative finance, genomic sequencing
 - Tests with LHC Atlas tag data

How To Learn More / Get Involved?

- LSST, including qserv
 - Check out lsst database trac at <http://dev.lsstcorp.org/trac/wiki/LSSTDatabase>
- XLDB
 - Attend XLDB4 (Oct 6-7 @SLAC)  **Open** conference starting this year
 - Read past XLDB reports <http://www-conf.slac.stanford.edu/xldb>
 - Share your use cases, join the community
- SciDB  **1st public release (alpha) next month**
 - Check out <http://scidb.org>
 - Follow through mailing list(s), on Twitter, soon LinkedIn, Facebook
 - Try it out
 - Attend community meeting (Oct 7 @SLAC)

Summary

- Exascale is closer than you think
- Shared-nothing clusters for extreme scale computing
- New techniques required for clustering-intensive, multi-d, uncertain data
- Solution providers are starting to address big science needs